Image-Based Walk-Through System for Large-Scale Scenes

Takaaki Endo[1], Akihiro Katayama[1], Hideyuki Tamura[1],
Michitaka Hirose[2], Tomohiro Tanikawa[2], and Makoto Saito[2]
[1]Mixed Reality Systems Laboratory Inc. 6-145 Hanasakicho, Nishi-ku,
Yokohama 220-0022 Japan
[2]University of Tokyo, 7-3-1 Hongo, Bunkyo-ku Tokyo 113-8656 Japan

Abstract. This paper describes the Image-Based Walk-Through System in which people walk-through a town which is historically valuable such as Shirakawa-go stored in a computer. This system is composed of two methods. The panorama-based method changes panoramic images according to the observer's point of view. The panoramic images used in this method is made by integrating images taken by the data capturing system. The morphing method generates images from arbitrary viewpoints from the original images. We can efficiently produce an image-based walk-through system in which people can move and look around rather freely by applying these two methods together.

1. Introduction

Some attempts are being done to digitize scenes of a historically valuable town such as Shirakawa-go which has been designated as one of the world heritage, and to make it possible to walk-through this digital town. These attempts are meaningful not only from the point of protecting a world heritage but also making the protection activities popular to various people.

Generally, 3-D geometrical model rendering is used to generate a large-scale virtual environment such as a town [1][2]. However, this methodology can not produce photo-realistic images under real-time conditions. Recovering geometrical models are not necessarily required in order to render only the external appearance of buildings. It is possible to produce images to be shown to an observer from two dimensional images without explicitly recovering geometrical models. This method is called Image-Based Rendering (IBR) and is one reasonable method to render virtual environments.

As such a method, a panorama-based method has been developed, which changes the panoramic images according to the observer's point of view [3]. This method allows the observer to see image sequences taken by the cameraman and look around freely, and makes it simple to produce a virtual space. But switching panoramic images cannot provide an image at a position apart from the camera position.

Thus we have applied a morphing method to provide an image at any position. The morphing method has the advantage of freely changing the point of view. Moreover, since the morphing method can produce intermediate images even if the original images are not so close to each other, we can reduce the amount of data required when only the panorama-based method is used. But it is not simple to produce a virtual space using the morphing method because the user have to specify corresponding regions of the original images by hand.

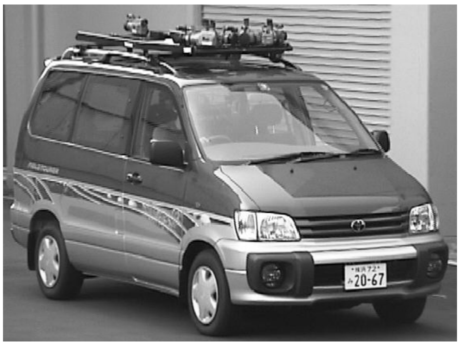

Figure 1: Data capturing system

In this paper, we introduce the image-based walk-through system in which an observer can move and look around rather freely by combining the advantages of these two methods.

2. Data Capturing System

The images of a broad area of a town are taken using cameras placed on an automobile [3] (Figure 1). The automobile is equipped with eight video cameras, a GPS (Global Positioning System), a gyroscope and a geomagnetic sensor in order to take images all around the automobile with position and orientation information.

3. Image Reproduction Method

As previously mentioned, we have developed a panorama-based method and a morphing method as an image reproduction method, but both have some defects. In this section, we discuss about the improvement of both methods in detail.

3.1 Panorama-Based Method

The panoramic image is made by projecting each image taken by the eight cameras onto a cylindrical surface and then aligning them horizontally after compensating distortion. This system can interactively provide images in any angle at any point by rendering the appropriate portion of the panoramic image according to the observer's request.

This system reads an array of panoramic images into the main memory in order to switch images in real time. The 256 Mbytes memory of our ONYX2 Graphics Workstation restricts the range of motion to about 20 meters. It is possible to extend the motion range by reading the images to be rendered directly from the hard disk, but we cannot achieve rendering speed faster than 2 frames per second.

Our new system uses the following approaches to make it possible to render images directly read from the hard disk in real time.

[Utilizing RAID Disk]

In order to speed up the disk access, we have used a RAID disk which can transfer data from the disk to the main memory at 100 Mbps.

[Shrinking the Size of the Images]

In order to speed up the image reading, we have used images halved in width and height. The size of the halved panoramic image is 2,136 x 243 pixels (approx. 1.4 Mbytes).

[Utilizing Multi-Threading Technology]

By making the reading and rendering processes parallel, we can reduce waiting time of the CPU at the time of disk access and make the image updating rate a little faster. In order to make the reading and rendering processes parallel, it is necessary to read the image to be rendered in the next step. Therefore, our system reads both previous and next panoramic images into the memory so that the system can render images already read into the memory no matter if the observer moves forward or backward.

These approaches speed up the rendering to about 10 frames per second.

3.2 Morphing Method

Since switching panoramic images cannot provide an image at a position and angle apart from the camera position, we have applied a morphing method to provide an image at any position in any angle.

First, images are divided into regions by specifying the corresponding region of 3 original images. Then, the observer can move freely inside the triangle area (whose nodes are the viewpoints of the 3 original images), by interpolating position and color among the corresponding regions of each original image.

Since the simple linear interpolation generates distortion in the image, we have developed a distortion free interpolation algorithm based on the View Morphing method [4]. This algorithm is as follows.

(Step 1) obtain an interpolation parameter from the coordinate positions of the viewpoint and the cameras of the three images.

If the coordinate position of the viewpoint is (X_e, Y_e, X_0) and the coordinate positions of the cameras are (X_n, Y_n, Z_0) (where n = 1, 2, 3), the weight parameter w_n (where n = 1, 2, 3) can be obtained from equation 1.

$$\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} X_1 & X_2 & X_3 \\ Y_1 & Y_2 & Y_3 \\ 1 & 1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} X_e \\ Y_e \\ 1 \end{bmatrix} \tag{1}$$

If the interpolation parameters s and t are as shown in figure 2, equation 2 represents the relationship of the parameters s and t. Parameters s and t can be obtained from this equation.

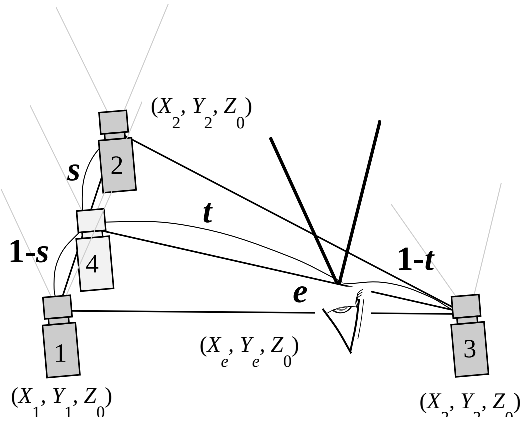

Figure 2: Interpolation parameters

$$s = \frac{w_1}{1 - w_3} \qquad t = w_1 \tag{2}$$

(Step 2) rotate the image so that the optical axes of I_1 and I_2 are parallel and the axes and the baseline are perpendicular.

The equation 3 represents the conversion from the coordinate point (x, y) in the original image to the coordinate point (x', y') in the rotated image.

$$x' = f\tan(\phi - \theta) \qquad y' = \frac{\cos\phi}{\cos(\phi - \theta)} \tag{3}$$

Where θ is the rotation angle, $f = W/2\tan(\alpha/2)$ (α is the image angle and W is the number of pixels in the horizontal direction of the image) and $\phi = \tan^{-1}(x/f)$. Note that the origin of the image is set at its center.

(Step 3) linearly interpolate the corresponding coordinate points in the rotated images of I_1 and I_2 using the ratio of s:1-s (equation 4).

$$\begin{bmatrix} x_4 \\ y_4 \end{bmatrix} = s\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + (1-s)\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} \tag{4}$$

Where (x_1, y_1) is the coordinate position of the point in the image I_1, (x_2, y_2) is the coordinate position of the point in the image I_2, and (x_4, y_4) is the coordinate position of the point in the image I_4 which is obtained by linearly interpolating camera positions in the I_I and I_2 images using the ratio of s:1-s.

(Step 4) rotate the image so that the optical axes of I_3 and I_4 are parallel and the axes and the baseline are perpendicular.

This step applies the same rotation conversion to the coordinate point (x_4, y_4) as the **step 2** above.

(Step 5) linearly interpolate the corresponding coordinate points in the rotated images of I_3 and I_4 using the ratio of t:1-t (equation 5).

$$\begin{bmatrix} x_e \\ y_e \end{bmatrix} = t\begin{bmatrix} x_3 \\ y_3 \end{bmatrix} + (1-t)\begin{bmatrix} x_4 \\ y_4 \end{bmatrix} \tag{5}$$

Where (x_3, y_3), (x_4, y_4), (x_e, y_e) are coordinate positions of the points in images I_3, I_4 and I_e.

(Step 6) rotate the image I_e so that it coincides with the viewing direction of the observer.

This step applies the same rotation conversion to the coordinate point (x_e, y_e) as the **step 2** above.

Note that it is impossible to rotate images so that the optical axes of the images are parallel and the axes and the baseline are perpendicular if the center of one camera is within the viewing area of one of other cameras. In this case we can not compensate the distortion in the method shown above. However, we can compensate the distortion by rotating the images so that the optical axes of the images I_A and I_B coincides with each other and then applying the following equation.

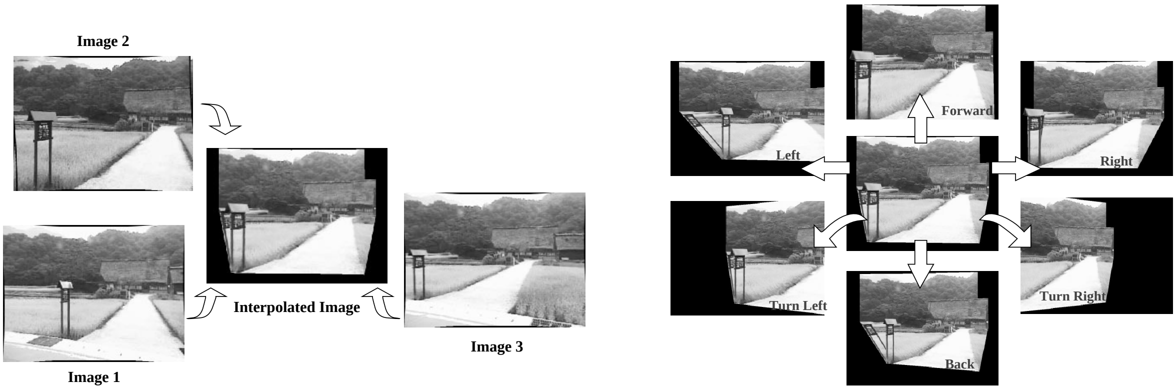

Figure 3: Morphing method

$$x_C = \frac{x_A x_B}{(1-u)x_A + u x_B}$$

$$y_C = \frac{y_A y_B}{(1-u)y_A + u y_B} \qquad (6)$$

Where $(x_A,\ y_A)$ and $(x_B,\ y_B)$ are coordinate positions of the point in the images I_A and I_B, and $(x_C,\ y_C)$ is a coordinate position of the point in the image I_C which is obtained by linearly interpolating camera positions in the I_A and I_B images using the ratio of u:1-u.

Figure 3 shows a result of the morphing method applied to the images captured at Shirakawa-go. We can also extend the movable area and let the observer look around by switching three original images according to the viewpoint of the observer.

4. Image-Based Walk-Through System

We have built a system where images of a town are projected on three screens and an observer can walk-through the virtual space by using a joystick (figure 4).

The observer uses a force feedback joystick (Impulse Engine) to move and look around the virtual space. The information of the joystick is taken into a PC and then sent to ONYX2 through the network. The ONYX2 workstation reads the information of the joystick from the shared memory and according to the information, it renders images by reading them from the RAID disk into the main memory.

Three directional images are rendered in a single image as shown in figure 5. The front view is at the upper left. The view of 60 degrees to the right is at the upper right and the view of 60 degrees to the left is at the lower left of the image. These images are separated through three digital scan converters into the images for each screen and then send to projectors. The projectors casts these separated images onto corresponding screens so that the observer can see scenes of 180 degrees and can move and look around using the joystick (figure 6).

The observer uses the joystick to switch the view so he/she can move back and forth by inclining the stick back and forth, and move left and right by inclining the stick to left and right while pressing a button. When the observer inclines the stick to left and right, the scene is switched as though the observer is looking around him/her. Another button is used to display a map in which an arrow shows the position and direction of the observer.

The system switches between the panorama-based method and the morphing method when

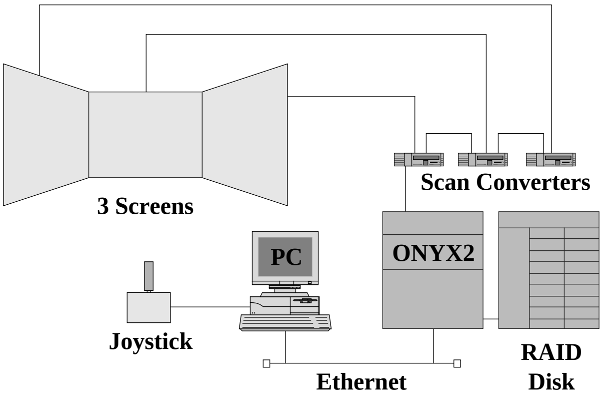

Figure 4: An overview of the system

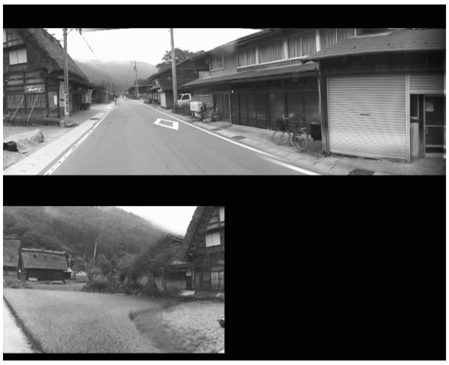

Figure 5: Three directional images

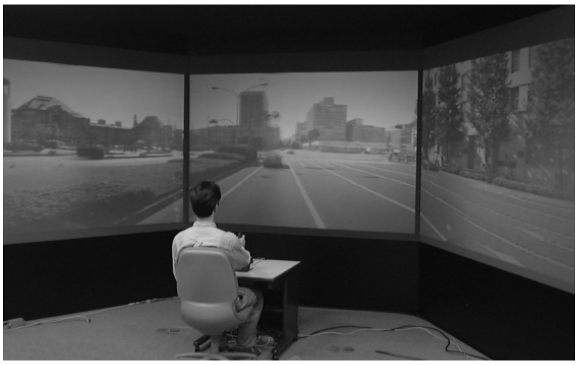

Figure 6: Images of a town projected on the three screens

the observer comes to a fixed position in the virtual environment and press the button. In morphing mode, the observer can move around an area of about 6 meters by 4 meters. Note that the panoramic images are read directly from the disk but the images for the morphing method are read into the main memory beforehand.

This system allows the observer to move around an area of approximately 800 x 200 meters. The number of panoramic images used in this system is about 5000 and its total size is 7 Gbytes.

5. Conclusion

This paper describes a new approach to digitize a town such as Shirakawa-go and to walk-through this digital town. In this system, we found that it is possible to let an observer to walk-through a broad photo-realistic virtual space rather freely by combining the panorama-based method and the morphing method.

We are going to examine a method to switch images more smoothly and to make the movable area presented by the morphing method broader.

References

[1] ModelCity™ Philadelphia: http://www.bentley.com/modelcity/
[2] CyberCity Berlin: http://www.echtzeit.de/products/CC.html
[3] M.Hirose, S.Watanabe, and T.Endo: "Generation of wide-range virtual spaces using photographic images," Proc. VRAIS '98, pp. 234-241 (1998).
[4] S.M.Seitz and C.R.Dyer: "View morphing," Proc. SIGGRAPH '96, pp.21-30 (1996).